3 Cyrix III BUS INTERFACE

The signals used in the Cyrix III CPU bus interface are described in this chapter. Figure 3-1 shows the signal directions and groups the signals for later description. Individual signal are described in Table 3-1 (Page 3-120).

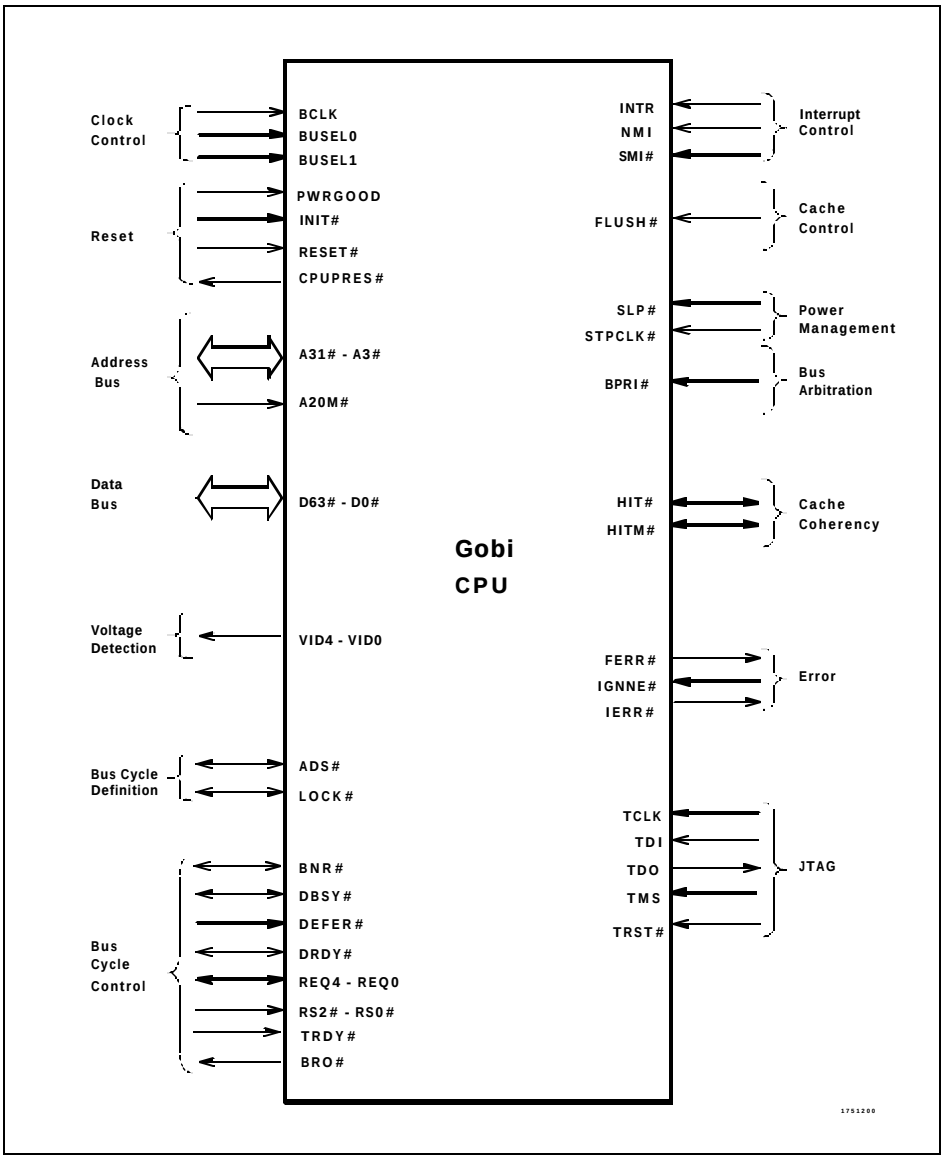

Figure 3-1. Cyrix III CPU Functional Signal Groupings

3.1 Signal Description Table

The Signal Summary Table (Table 3-1) describes the signals in their active state unless otherwise mentioned. Signals ending with a "#" character are active low.

Table 3-1. Cyrix III CPU Signals Sorted by Signal Name

Pin Name	Description	I/O	Clock
A[31 -3]#	The Address Bus provides addresses for physical memory and external I/O devices. During cache inquiry cycles, A31#-A3# are used as inputs to perform snoop cycles.	I/O (GTL+)	BCLK
A20M#	A20 Mask causes the CPU to mask (force to 0) the A20 address bit when driving the external address bus or performing an internal cache access. A20M# is provided to emulate the 1 MByte address wrap-around that occurs on the 8086. Snoop addressing is not affected.	I (2.5V)	Asynch
ADS#	Address Strobe begins a memory/I/O cycle and indicates the address bus (A31#-A3#) and transaction request signals (REQ#) are valid.	I/O (GTL+)	BCLK
BCLK	Bus Clock provides the fundamental timing for the Cyrix III CPU. The frequency of the Cyrix III CPU input clock determines the operating frequency of the CPU's bus. External timing is defined referenced to the rising edge of CLK.	I (2.5V)	--
BNR#	Block Next Request signals a bus stall by a bus agent unable to accept new transactions.	I/O (GTL+)	BCLK
BPRI#	Priority Agent Bus Request arbitrates for ownership of the system bus.	I (GTL+)	BCLK
BSEL[0 - 1]	Bus Selection Bus provides system bus frequency data to the CPU.	I (GTL+)	BCLK
BR0#	**Bus Request** Always asserted since Cyrix III supports only uni-processing.	Ground	None
CPUPRES#	CPU Present provides a ground to allow the motherboard to detect the cpu	Ground	None
D[63 - 0]#	Data Bus signals are bi-directional signals which provide the data path between the Cyrix III CPU and external memory and I/O devices. The data bus driver must assert DRDY# to indicate valid data transfer.	I/O (GTL+)	BCLK
DBSY#	Data Bus Busy is asserted by the data bus driver to indicate data bus is in use.	I/O (GTL+)	BCLK
DEFER#	Defer is asserted by target agent (e.g., north bridge) and indicates the transaction cannot be guaranteed as an in-order completion.	I (GTL+)	BCLK
DRDY#	Data Ready is asserted by data driver to indicate that a valid signal is on the data bus.	I/O (GTL+)	BCLK
FERR#	FPU Error Status indicates an unmasked floating point error has occurred. FERR# is asserted during execution of the FPU instruction that caused the error.	O (2.5V)	Asynch

Table 3-1. Cyrix III CPU Signals Sorted by Signal Name

Pin Name	Description	I/O	Clock
FLUSH#	Flush Internal Caches writing back all data in the modified state.	I (2.5V)	Asynch
HIT#	Snoop Hit indicates that the current cache inquiry address has been found in the cache (exclusive or shared states).	I/O (GTL+)	BCLK
HITM#	Snoop Hit Modified indicates that the current cache inquiry address has been found in the cache and dirty data exists in the cache line (modified state).	I/O (GTL+)	BCLK
IERR#	**Internal Error** Tied to a pull-up resistor. Never occurs.	Pull-up.	None
IGNNE#	Ignore Numeric Error forces the Cyrix III CPU to ignore any pending unmasked FPU errors and allows continued execution of floating point instructions.	I (2.5V)	Asynch
INIT#	Initialization resets integer registers and does not effect internal cache or floating point registers.	I (2.5V)	Asynch
INTR	Maskable Interrupt	I (2.5V)	Asynch
NMI	Non-Maskable Interrupt	I (2.5V)	Asynch
LOCK#	Lock Status is used by the CPU to signal to the target that the operation is atomic.	I/O (GTL+)	BCLK
PWRGOOD	Power Good indicates to the CPU that clocks and power supplies are stable and in specification.	I (2.5V)	Asynch
REQ4# - REQ0#	Request Command is asserted by bus driver to define current transaction type.	I/O (GTL+)	BCLK
RESET#	Resets the processor and invalidates internal cache without writing back.	I (GTL+)	BCLK
RS[2 - 0]#	Response Status signals the completion status of the current transaction when the CPU is the response agent.	I (GTL+)	BCLK
SLP#	Sleep, when asserted in the stop grant state, causes the CPU to enter the sleep state.	I (2.5V)	Asynch
SMI#	System Management (SMM) Interrupt forces the processor to save the CPU state to the top of SMM memory and to begin execution of the SMI service routine at the beginning of the defined SMM memory space. An SMI is a higher-priority interrupt than an NMI.	I (2.5V)	Asynch
STPCLK#	Stop Clock causes the CPU to enter the stop grant state.	I (2.5V)	Asynch
TCLK	Test Clock (JTAG) is the clock input used by the Cyrix III CPU's boundary scan (JTAG) test logic. (Called TCK by Intel)	I (2.5V)	--
TDI	Test Data In (JTAG) is the serial data input used by the Cyrix III CPU's boundary scan (JTAG) test logic.	I (2.5V)	TCLK
TDO	Test Data Out (JTAG) is the serial data output used by the Cyrix III CPU's boundary scan (JTAG) test logic.	O (2.5V)	TCLK
TMS	Test Mode Select (JTAG) is the control input used by the Cyrix III CPU's boundary scan (JTAG) test logic.	I (2.5V)	TCLK

Signal Description Table

Table 3-1. Cyrix III CPU Signals Sorted by Signal Name

Pin Name	Description	I/O	Clock
TRDY#	Target Ready indicates that the target is ready to receive a write or write-back transfer from the CPU.	I (GTL+)	BCLK
TRST#	Test Mode Reset (JTAG) initializes the Cyrix III CPU's boundary scan (JTAG) test logic.	I (2.5V)	Asynch
VID[4 - 0]	Voltage Identification Bus informs the regulator system on the motherboard of the CPU Core voltage requirements.	O (2.5V)	Asynch

Table 3-2. Intel® Celeron™ Signals Not Supported by Cyrix® Cyrix III CPU

Pin Name	Description	Reason
BP[3:2]#	Breakpoint.	Debug extension not supported.
BPM[1:0]#	Breakpoint and performance monitor.	Debug extension not supported.
BR[0]#	Bus request.	Multiprocessing not supported.
IERR#	Internal error.	Internal error detection not supported.
PICCLK	Advanced Programmable Interrupt Controller (APIC) clock.	Multiprocessor mode and therefore the APIC bus is not required.
PICD[1:0]	Advanced Programmable Interrupt Controller (APIC) data.	
PRDY#	Probe ready.	Intel debug tools are not supported.
PREQ#	Probe request.	
THERMTRIP#	Thermal Sensor.	Over temperature trip signal is not supported, The CPU does have a thermal diode but supports the other temperature related signals.

3.2 Signal Descriptions

The following paragraphs provide additional information about the Cyrix III CPU signals. For ease of this discussion, the signals are divided into 16 functional groups as illustrated in Figure 3-1 (Page 3-119).

3.2.1 Bus Clock

The Bus Clock Input (BCLK) signal, supplied by the system, is the timing reference used by the Cyrix III CPU bus interface. All external timing parameters are defined with respect to the BCLK rising edge. The BCLK signal enters the Cyrix III CPU where it is multiplied to produce the Cyrix III CPU internal clock signal. During power on, the CLK signal must be running even if CLK does not meet AC specifications.

3.2.2 Acquiring the CPU/Bus Clock Ratio

The CPU/Bus clock ratio is acquired from the group of input signals listed in Table 3-3 (Page 3-125) during reset. Unlike the Intel Celeron processor which hardwires CPU core clock ratios, the Cyrix III processor uses this original Pentium II method for setting the core clock ratio.

The Cyrix III CPU samples NMI, INTR, A20M# and IGNNE# while RESET# is asserted (low) and latches the values at the rising edge of RESET#. These signals must be stable for 1ms prior to the rising edge of RESET# to ensure proper operation.

There is a pin called external ratio pin, EXTRATIO#, which must be low for the clock ratio jumpers to work. This pin defaults to high in the cpu, so if not hooked up, or hooked up to Vcc, the clock ratio jumpers will not work and the cpu default will boot up.

Clock ratios under 2.5:1 and selections identified as not applicable (N/A) are not supported. All clock ratios and frequencies listed may not be available.

A BIOS method for setting the clock ratio is also available. See chapter 2 for the configuration register programming method of setting the clock multiplier. When using the BIOS method only, no jumpers, then EXTRATIO# must be tied high, so that invalid information on the jumper pins is not latched into the cpu before BIOS programming can take place.

Thus there are three ways to set the bus clock multiplier: With board jumpers, with BIOS, or simply using cpu default.

Table 3-3. CPU/Bus Frequency Ratio Encoding

CPU/BUS Ratio	NMI	INTR	A20M#	IGNNE#	Core Block (MHz)		
					66 MHz Bus	100 MHz Bus	133 MHz Bus
2.5	L	H	L	L			333
3.0	L	L	L	H		300	400
3.5	L	H	L	H		350	466
4.0	L	L	H	L		400	
4.5	L	H	H	L	300	450	
5.0	L	L	H	H	333	500	
5.5	L	H	H	H	366		
6.0	H	L	L	L	400		
6.5	H	H	L	L	433		
7.0	H	L	L	H	466		
7.5	H	H	L	H	500		

3.2.3 Providing CPU Voltage Information

The CPU supply voltage requirements are provided by the Voltage Identification Bus (VID) Bus. This bus consists of five CPU pins VID[4 - 0]. The encoding of these pins is listed in Table 3-4.

Note that an “L” in the table indicates the output pin is switched to ground, and a “H” indicates that the CPU pin is open. The VID bus identifies the same voltages on the Celeron from 1.80 to 2.05 volts inclusive. Other encodings are reserved.

Table 3-4. Voltage Identification Bus

Selected CPU Core Voltage in Volts	VID4	VID3	VID2	VID1	VID0
1.80	L	L	H	L	H
1.85	L	L	H	L	L
1.90	L	L	L	H	H
1.95	L	L	L	H	L
2.00	L	L	L	L	H
2.05	L	L	L	L	L
2.10	H	H	H	H	L
2.20	H	H	H	L	H

3.2.4 Acquiring Bus Speed Information

The CPU acquires the bus by sampling pins BSEL0 and BSEL1 pins during reset. The bus speed information is used to setup the internal clocks within the CPU and tune the I/O bus interface. The Cyrix III processor currently supports 66 MHz, 100 MHz and 133 MHz bus frequencies.

Table 3-5. Bus Speed Signaling

BSEL1	BSEL0	Bus Frequency (MHz)
X	L	66
L	H	100
H	H	133

3.2.5 Reset Control

The Cyrix III CPU output signals are initialized to their reset states during the CPU reset sequence.

Asserting RESET# suspends all operations in progress and places the Cyrix III CPU in a reset state. RESET# can be asserted asynchronously but must be de-asserted synchronously from the clock.

On system power-up, RESET# must be held asserted for at least 1 millisecond after PWRGOOD, Vcc and CLK have reached specified DC and AC limits. This delay allows the CPU's clock circuit to stabilize and guarantees proper completion of the reset sequence.

During normal operation, RESET# must be asserted for at least 1 microsecond in order to guarantee the proper reset sequence is executed.

3.2.6 Address Bus

The Address Bus (A[31-3]#) lines provide the physical memory and external I/O device addresses. A[31-3]# are bi-directional signals used by the Cyrix III CPU to drive addresses to both memory devices and I/O devices. During cache inquiry cycles the Cyrix III CPU receives addresses from the system using signals A[31-3]#.

Using signals A[31-3]#, the Cyrix III CPU can address a 4-GByte memory address space. Using signals A[15-3]#, the Cyrix III CPU can address a 64-KByte I/O space through the processor's I/O ports. During I/O accesses, signals A[31-16]# are driven low.

Address Bit 20 Mask (A20M#) is an active low input which causes the Cyrix III CPU to mask (force low) physical address bit 20 when driving the external address bus or when performing an internal cache access. Asserting A20M# emulates the 1 MByte address wrap-around that occurs on the 8086. The A20 signal is never masked during write-back cycles, inquiry cycles, system management address space accesses or when paging is enabled, regardless of the state of the A20M# input.

3.2.7 Data Bus

Data Bus (D63#-D0#) lines carry, bi-directional signals between the Cyrix III CPU and the system (i.e., external memory and I/O devices). The data bus transfers data to the Cyrix III CPU during memory read, I/O read, and interrupt acknowledge cycles. Data is transferred from the Cyrix III CPU during memory and I/O write cycles.

Data setup and hold times must be met for correct read cycle operation. The data bus is driven only while a write cycle is active.

3.2.8 Bus Cycle Definition

A bus transaction is divided into six phases: arbitration phase, request phase, error phase, snoop phase, response phase, and data phase. Each phases is assigned a different set of processor pins.

3.2.8.1 Arbitration Phase

Three CPU pins are used during the arbitration phase:

- Priority Agent Bus Request (BPRI#)
- Block Next Request (BNR#)
- Lock (LOCK#).

When the chipset requires the bus, a request for bus ownership is generated by BPRI# on the chipset. After BPRI# is asserted, the chipset is assigned as the priority agent and will be granted ownership of the next available transaction.

If the P6 bus agent (e.g., CPU or north bridge) anticipates that the system resources, such as the address and data buffers, are going to be temporarily busy, BNR# can be asserted to delay the next transaction. During the bus stall no agent will be granted bus control.

During non-interruptible sequences of bus transactions, LOCK# is asserted to prevent other agents from accessing the bus. This process allows the processor to maintain control of the bus for a series of transactions.

3.2.8.2 Request Phase

During the Request phase the following signals are used:

- Address Strobe (ADS#)
- Request Bus (REQ[4-0]#)
- Address Bus (A[31-3]#)

The Cyrix III asserts ADS# when it is ready to begin the transaction. REQ[4-0]# and A[31-3]# are valid in the clock cycle and one clock after in which ADS# is asserted.

The REQ[4-0]# pins define the type of transaction according to Table 3-6.

Table 3-6. Transaction Types

REQ4#	REQ3#	REQ2#	REQ1#	REQ0#	Transaction Type
0	0	0	0	0	Deferred Reply
0	0	0	0	1	Reserved
0	1	0	0	0	Interrupt Acknowledge or Special Cycle
0	1	0	0	1	Reserved
1	0	0	0	0	I/O Read
1	0	0	0	1	I/O Write
1	1	0	0	x	Reserved
x	x	0	1	0	Memory Read and Invalidate
x	x	0	1	1	Reserved
x	x	1	0	0	Memory Code Read
x	x	1	1	0	Memory Data Read
x	x	1	0	1	Memory Write (no retry)
x	x	1	1	1	Memory Write (may retry)

3.2.8.3 Error Phase

The CPU is passive during Error Phase.

3.2.8.4 Snoop Phase

The snoop signals are:

- Cache Hit (HIT#)
- Cache Hit Modified (HITM#),
- Transaction Defer (DEFER).

HIT# and HITM# are used to maintain cache coherency. The snooped agent asserts HIT# when a snooped line is in the exclusive or shared state in the cache. HITM# is asserted when the line is in the modified state in the cache.

If HIT# and HITM# are asserted together, it indicates that the caching agent is not ready to indicate the snoop status and the snoop phase must be extended.

DEFER# indicates that the transaction cannot be guaranteed to complete in order. The response agent will either be issuing a retry or defer transaction in the response phase.

3.2.8.5 Response Phase

The response pins are:

- Response Status (RS2-0)
- Target Ready (TRDY#).

RS[2-0] provides the response status from the response agent. The status of the response to the transaction request according to RS is listed in the following table:

Table 3-7. Response Type

RS2#	RS1#	RS0#	Response Type
0	0	0	Idle. This is the default state when no response is being delivered.
0	0	1	Retry. Request agent must try the transaction later.
0	1	0	Deferred. The response agent will service the request later.
0	1	1	Reserved
1	0	0	Hard Failure. The response agent cannot service the request.
1	0	1	No Data. No data is needed from the response agent, as in a write for example.
1	1	0	Implicit Writeback. The transaction resulted in a hit on a modified line, and the snoop agent must supply the data. The response agent receives the writeback.
1	1	1	Normal Data. Data will be sent by the response agent, as in a normal read.

The response agent asserts TRDY when it is ready to accept the written data or writeback data.

3.2.8.6 Data Phase

The data pins are:

- Data Bus Ready (DRDY#)
- Data Bus Busy (DBSY#)
- Data Bus (D[63-0]#).

The driving agent on the bus asserts DRDY# to indicate that the data being driven onto the data bus is valid. The bus driver must assert DRDY# for each bus clock in which there is valid data to be transferred. DRDY# can be deasserted to insert wait states between data transfers.

DBSY# is used to hold the bus during wait states before the first DRDY# and between consecutive DRDY#. DBSY# indicates to other bus agent that the data bus is in use even though there is no valid data currently on the bus.

DBSY# need not be asserted for single clock data transfers with no wait states.

3.2.9 Interrupt Control

The interrupt control signals (INTR, NMI, SMI#) allow the execution of the current instruction stream to be interrupted and suspended.

Maskable Interrupt Request (INTR) is an active high level-sensitive input which causes the processor to suspend execution of the current instruction stream and begin execution of an interrupt service routine. The INTR input can be masked (ignored) through the IF bit in the Flags Register.

When not masked, the Cyrix III CPU responds to the INTR input by performing an interrupt acknowledge bus cycle. During the interrupt acknowledge cycle, the Cyrix III CPU reads the interrupt vector (an 8-bit value), from the data bus. The 8-bit interrupt vector indicates the interrupt level that caused generation of the

INTR and is used by the CPU to determine the beginning address of the interrupt service routine. To assure recognition of the INTR request, INTR must remain active until the start of the interrupt acknowledge cycle.

Non-Maskable Interrupt Request (NMI) is a rising edge sensitive input which causes the processor to suspend execution of the current instruction stream and begin execution of an NMI interrupt service routine. The NMI interrupt cannot be masked by the IF bit in the Flags Register. Asserting NMI causes an interrupt which internally supplies interrupt vector 2h to the CPU core. Therefore, external interrupt acknowledge cycles are not issued.

Once NMI processing has started, no additional NMIs are processed until an IRET instruction is executed, typically at the end of the NMI service routine. If NMI is re-asserted prior to execution of the IRET, one and only one NMI rising edge is stored and then processed after execution of the next IRET.

System Management Interrupt Request (SMI#) is an interrupt input with higher priority than the NMI input. Asserting SMI# forces the processor to save the CPU state to SMM memory and to begin execution of the SMI service routine.

SMI# behaves one of two ways depending on the SMM mode of the CPU.

In SL-compatible mode SMI# is a falling edge sensitive input and is sampled on every rising edge of the processor input clock. Once SMI# servicing has started, no additional SMI# interrupts are processed until a RSM instruction is executed. If SMI# is reasserted prior to execution of a RSM instruction, one and only one SMI# falling edge is stored and then processed after execution of the next RSM.

In Cyrix enhanced SMM mode, SMI# is level sensitive. As a level sensitive input, software can process all SMI interrupts until all sources in the chipset have cleared.

3.2.10 FPU Error Interface

The FPU interface signals FERR# and IGNNE# are used to control error reporting for the on-chip floating point unit. These signals are typically used for a PC-compatible system implementation. For other applications, FPU errors are reported to the Cyrix III CPU core through an internal interface.

Floating Point Error Status (FERR#) is an active low output asserted by the Cyrix III CPU when an unmasked floating point error occurs. FERR# is asserted during execution of the FPU instruction that caused the error.

Ignore Numeric Error (IGNNE#) is an active low input which forces the Cyrix III CPU to ignore any pending unmasked FPU errors and allows continued execution of floating point instructions. When IGNNE# is not asserted and an unmasked FPU error is pending, the Cyrix III CPU only executes the following floating point instructions: FNCLEX, FNINIT, FNSAVE, FNSTCW, FNSTENV, and FNSTSW#. IGNNE# is ignored when the NE bit in CR0 is set to a 1.